

Projet Tutoré AP2

Shoot Them Up

Rapport

Guillaume ISSELIN
Maxime VOLFOVSKY

31 mai 2013

IUT Bordeaux 1
Département Informatique
Année Universitaire 2012-2013

Enseignant : M. BILLAUD

Sommaire

1	Introduction	3
2	Mode d'emploi	4
2.1	Menu	4
2.2	Règles du jeu	4
2.2.1	Déplacement du joueur	5
2.2.2	Tirer	5
2.2.3	Jauge de Vie	5
2.2.4	Vies	6
2.2.5	Niveaux	6
2.3	Améliorations (PowerUps)	6
2.3.1	100k	6
2.3.2	Coeur	6
2.3.3	Protection	6
2.3.4	Bombe	7
2.3.5	Jauge de Vie + ...	7
3	Diagramme de classes UML (partie Modèle)	8
4	Algorithmes	9
4.1	Génération aléatoire des ennemis	9
4.2	Déplacement des ennemis	9
4.3	Déplacement des tirs	10
5	Conclusion	10
6	Crédits et remerciements	11
7	Licence d'utilisation	12
8	Annexe	12
8.1	Code source C++	12

1 Introduction

Au cours du premier semestre, la réalisation du projet "Duck Hunt", nous avait tellement passionné, qu'au début du second semestre, la première chose qui nous est venue à l'esprit fut : Mais que sera donc le projet de programmation orientée objet à développer ? (si, sérieusement !)

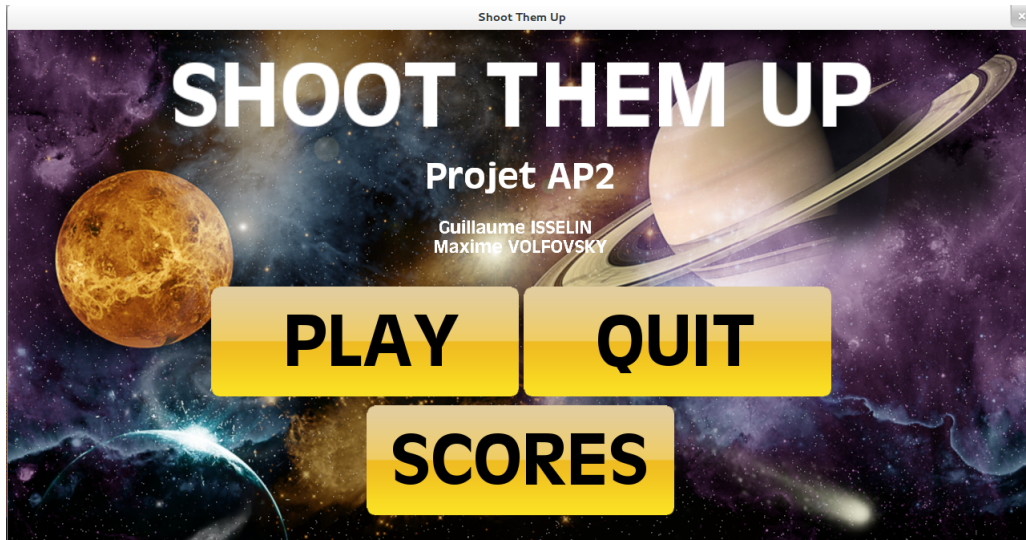
Le 25 février 2013, le jour tant attendu arriva : la présentation du projet. Nous avons ainsi appris que nous devions développer un *Shoot Them Up*.

Sur le coup, nous avons été emballé par le concept du jeu, même si un de nous deux (Guillaume) n'avait jamais entendu parler de *Shoot Them Up*.

Ainsi, nous allons vous présenter notre *Shoot*. Tout d'abord, nous décrirons dans un premier temps comment utiliser notre jeu. Ensuite, nous expliquerons l'architecture du jeu, en utilisant un diagramme de classes UML. Puis, nous décrirons quelques algorithmes que nous avons jugés intéressants d'expliquer. Enfin, nous concluons ce rapport, en indiquant ce que le développement de ce projet nous a apporté.

2 Mode d'emploi

2.1 Menu



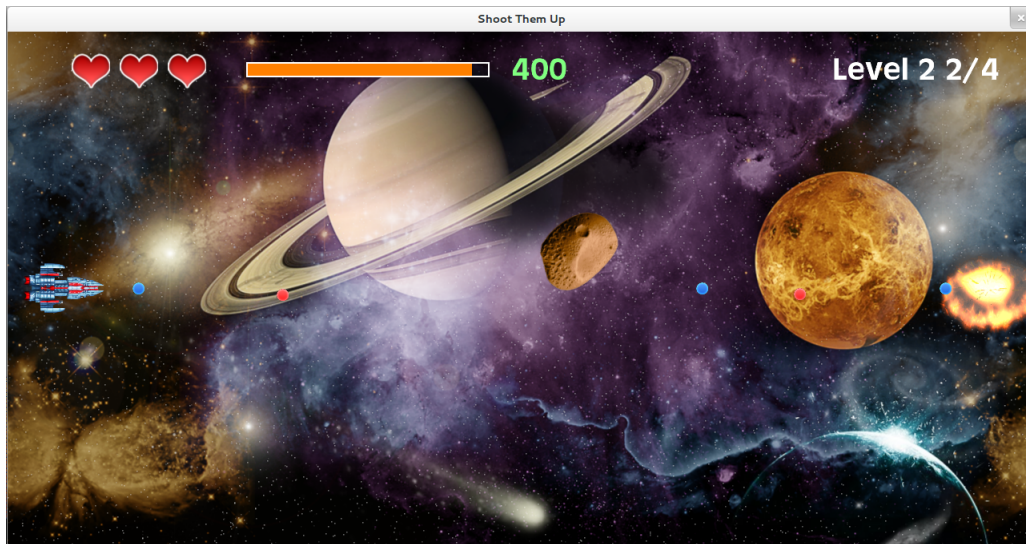
Au démarrage du jeu, vous arrivez à l'écran de menu. Cet écran vous permet de :

- Démarrer une partie
- Quitter le jeu
- Afficher les meilleurs scores
- Activer les modes de jeu secrets ¹

2.2 Règles du jeu

Le but de ce jeu est simple : tuer le plus d'ennemis possible, pour marquer le plus de points possible. Cela peut paraître simple sur le papier, mais vous serez limité dans votre action par plusieurs facteurs. Tout d'abord, au fur et à mesure des niveaux, des ennemis coriaces tirant plusieurs fois d'un coup apparaîtront de plus en plus fréquemment. Le fait que vous possédiez une jauge de santé, un nombre de vies limité, et qu'il y ait des astéroïdes pouvant vous barrer la route, pimentera encore plus le jeu. Nous allons décrire toutes les actions et événements qui peuvent se passer dans le jeu.

1. Enfin, après avoir vu le code source, ces modes ne le seront plus tout à fait...Il faut utiliser les touches **H** et **C**



2.2.1 Déplacement du joueur

Vous devez déplacer votre vaisseau pour viser les ennemis, éviter les tirs des ennemis, et attraper des Power-Ups.

Pour cela, utilisez les **flèches directionnelles** de votre clavier pour vous déplacer dans la direction que vous souhaitez.

2.2.2 Tirer

Les tirs **bleus** correspondent à vos propres tirs et les tirs **rouges** à ceux des ennemis.

Pour utiliser un tir basique, maintenez la touche **Espace** enfoncée.

Vous disposez également de tirs bonus, utilisables à tout moment :

- **Enhanced Shoot 3000** : permet de tirer **3** tirs d'un coup, chacun partant dans une direction différente.

Pour l'utiliser, maintenez la touche **E** de votre clavier enfoncée.

- **Amazing Shoot 5000** : permet de tirer **5** tirs d'un coup, chacun partant dans une direction différente.

Pour l'utiliser, maintenez la touche **A** de votre clavier enfoncée.

Attention ! L'utilisation de ces tirs "bonus" n'est tout de même pas "gratuite" : il vous en coûtera quelques points de vie à chaque utilisation. Faites donc très attention lorsque vous utilisez ces tirs bonus.

2.2.3 Jauge de Vie

La jauge de vie correspond à la barre orange en haut de la fenêtre du jeu.

Dès que votre jauge de vie est à 0, vous perdez une vie.

Votre jauge de vie diminue à chaque fois que vous êtes touché par un tir ennemi, ou dès que vous utilisez un tir "bonus".

2.2.4 Vies



Vous possédez, au début du jeu, 3 vies, correspondant à 3 "chances" d'aller le plus loin dans votre quête de l'espace inter-sidéral. Vous perdrez une vie à chaque fois que votre jauge de vie descendra à 0. Vous perdrez également une vie si jamais vous entrez en collision avec un vaisseau ennemi.

Si vous n'avez plus de vies, vous perdrez la partie.

2.2.5 Niveaux

Lorsque vous avez tué un certain nombre d'ennemis, un message vous informera que vous pouvez passer au niveau suivant.

Ce nombre d'ennemis à tuer est indiqué en haut à droite de la fenêtre du jeu.

Pour votre information, sachez que les niveaux supérieurs à 11, sont **EXTRÊMEMENT DIFFICILES**.

Autre chose : le nombre de niveaux dans le jeu est infini. Tous les ennemis sont générés aléatoirement, au fur et à mesure des niveaux la fréquence d'apparition est plus importante, et de nouveaux types d'ennemis apparaissent.

2.3 Améliorations (PowerUps)

Si vous êtes une personne chanceuse, alors vous aurez, peut-être, l'immense privilège de voir apparaître dans le jeu, des Power-Ups permettant de progresser plus loin, et de faire un bien meilleur score.

Lors de la rencontre avec un Power-Up, il vous suffit d'entrer en collision avec pour l'activer.

2.3.1 100k



Ce Power-Up ajoute immédiatement 100 000 Points à votre score actuel. De quoi vous aider à battre vos meilleurs scores !

2.3.2 Coeur



Ce Power-Up vous permet de regagner 1 vie, et de réinitialiser votre jauge de vie au maximum.

2.3.3 Protection



Ce Power-Up vous protège de toutes les attaques pendant 5 secondes.

2.3.4 Bombe



Ce Power-Up vous permet de tuer tous les ennemis présents à l'écran.

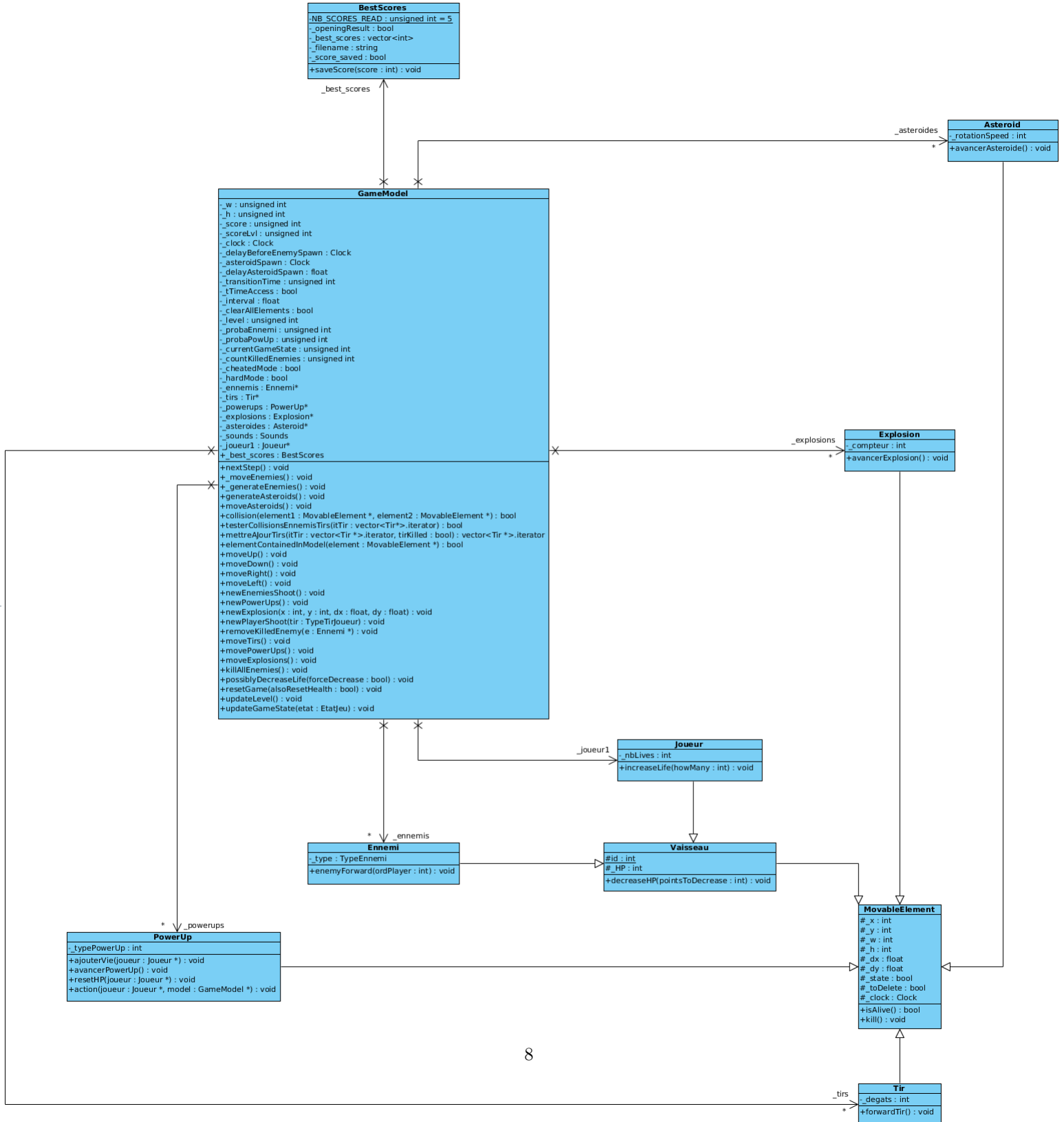
Attention : lors de l'utilisation de ce bonus, les ennemis détruits ne vous rapporteront pas de points, et ne seront pas comptabilisés dans le compteur d'ennemis tués. Ainsi, vous ne pouvez pas changer de niveau immédiatement après avoir utilisé un Power-Up Bombe

2.3.5 Jauge de Vie +



Ce Power-Up vous permet de régénérer votre jauge de vie au maximum.

3 Diagramme de classes UML (partie Modèle)



4 Algorithmes

4.1 Génération aléatoire des ennemis

Algorithme 1:

```
Action générerEnnemis()
si 5 secondes écoulées après début niveau alors
  si niveau >= 1 ET niveau < 3 alors
    On génère un ennemi facile
  sinon si niveau >= 3 ET niveau < 5 alors
    50% de chance de générer un ennemi FACILE
    50% de chance de générer un ennemi MOYEN
  sinon si niveau >= 5 ET niveau < 10 alors
    30% de chance de créer un ennemi FACILE
    30% de chance de créer un ennemi MOYEN
    40% de chance de créer un ennemi HARD
  sinon si niveau >= 10 alors
    20% de chance de créer un ennemi FACILE
    20% de chance de créer un ennemi MOYEN
    60% de chance de créer un ennemi HARD
```

4.2 Déplacement des ennemis

Algorithme 2:

```
Action déplacerEnnemis()
tant que NON vecteur ennemis parcouru faire
  si collision(vaisseau joueur, ennemi) alors
    Tuer ennemi
    Enlever vie au joueur
    Supprimer Ennemi du vecteur d'Ennemis
  sinon si NON collision(joueur, ennemi) ET ennemi dans écran alors
    si Ennemi_Y <= Joueur_Y alors
      Ennemi_Y ← Ennemi_Y + Ennemi_DY
    sinon si Ennemi_Y >= Joueur_Y alors
      Ennemi_Y ← Ennemi_Y - Ennemi_DY
    Ennemi_X ← Ennemi_X + Ennemi_DX
  sinon si ennemi est sorti de l'écran alors
    Tuer ennemi
    Supprimer Ennemi du vecteur d'Ennemis
```

4.3 Déplacement des tirs

Algorithme 3:

```
Action déplacerTirs()
tant que NON vecteur tirs parcouru faire
    si collision(joueur, tir) alors
        Enlever points de vie au joueur
        Supprimer Tir du vecteur de Tirs
    sinon si collision(ennemi, tir) alors
        Tuer ennemi Supprimer Tir du vecteur de Tirs
        Supprimer Ennemi du vecteur d'Ennemis
    sinon
        Tuer ennemi
        Supprimer Ennemi du vecteur d'Ennemis
```

5 Conclusion

Dans ce projet de fin de première année, nous avons pu mettre en pratique l'enseignement de Programmation Orientée Objet que nous avons reçu. Il nous a permis d'utiliser les notions d'héritage (avec toutes les classes dérivées des éléments mobiles), mais également les fonctions de la STL, avec entre autres les `vector`, une `map`, la fonction de tri `sort`...

La réalisation de ce projet s'est faite grâce à la bibliothèque graphique SFML qui est uniquement orientée objet. Cette bibliothèque permet une utilisation simple et complète d'éléments multimédias (sons, musiques, graphismes, fenêtrage), et est donc parfaite pour la réalisation du Shoot Them Up.

Au cours du projet, l'utilisation de Mercurial/Bitbucket nous a été d'une grande aide pour maintenir la cohérence dans les versions produites.

Nous en avons également profité pour changer d'IDE. Geany devenait difficile d'utilisation car le projet commençait à prendre de l'ampleur... Nous avons donc décidé d'utiliser **QtCreator**. Cet IDE a l'avantage d'intégrer directement le gestionnaire de versions Mercurial, et une interface avec **GDB**, entre autres. Il possède également de nombreuses petites fonctionnalités qui nous ont facilité le développement du jeu : génération automatique des accesseurs, mise à jour automatique de la déclaration et de l'implémentation d'une fonction, auto-complétion...

Ce changement nous a donc fait gagner beaucoup de temps lors de la réalisation du projet. Pour finir nous avons changé de compilateur (uniquement sur nos ordinateurs personnels), en utilisant **clang** à la place de **g++**. **clang** est compatible à 100% avec notre code, et a la particularité de compiler plus vite ; nous gagnons en effet 40% de temps par rapport à une compilation complète avec **g++**. Les messages d'erreur et warnings sont affichés en couleur et facilitent grandement la lecture des informations fournies par le compilateur.

6 Crédits et remerciements

La réalisation de ce jeu n'aurait sans doute pas été possible sans la participation de plusieurs personnes :

Tout d'abord, nous souhaitons remercier M. Billaud, pour tous ses conseils avisés, transmis toujours dans la bonne humeur.

Enfin, nous remercions Angélique Martinez, pour avoir créé l'arrière-plan du jeu.

Nous avons utilisé, dans ce jeu, plusieurs images extraites de sources différents :

- Feuille de sprites pour l'explosion : OpenGameArt.org
- Vaisseaux spatiaux : OpenGameArt.org / modifiés par G. Isselin
- Images power-ups : Guillaume Isselin

La musique utilisée dans le jeu est :

« *Blue Monday* » du groupe **New Order** (1987)

Les sons sont également extraits d'OpenGameArt.org, sauf le son se produisant lorsqu'on gagne une vie (soundfxcenter.com).

7 Licence d'utilisation



Vous êtes libre de :

partager — reproduire, distribuer et communiquer l'œuvre

remixer — adapter l'œuvre

Selon les conditions suivantes :

Attribution — Vous devez attribuer l'œuvre de la manière indiquée par l'auteur de l'œuvre ou le titulaire des droits (mais pas d'une manière qui suggérerait qu'ils vous approuvent, vous ou votre utilisation de l'œuvre).

Pas d'Utilisation Commerciale — Vous n'avez pas le droit d'utiliser cette œuvre à des fins commerciales.

Partage dans les Mêmes Conditions — Si vous modifiez, transformez ou adaptez cette œuvre, vous n'avez le droit de distribuer votre création que sous une licence identique ou similaire à celle-ci.

En comprenant bien que :

Renonciation — N'importe laquelle des conditions ci-dessus peut être levée si vous avez l'autorisation du titulaire de droits.

Domaine Public — Là où l'œuvre ou un quelconque de ses éléments est dans le domaine public selon le droit applicable, ce statut n'est en aucune façon affecté par la licence.

Autres droits — Les droits suivants ne sont en aucune manière affectés par la licence :

- Vos prérogatives issues des exceptions et limitations aux droits exclusifs ou fair use ;
- Les droits moraux de l'auteur ;
- Droits qu'autrui peut avoir soit sur l'œuvre elle-même soit sur la façon dont elle est utilisée, comme le droit à l'image ou les droits à la vie privée.

Remarque — A chaque réutilisation ou distribution de cette œuvre, vous devez faire apparaître clairement au public la licence selon laquelle elle est mise à disposition.

8 Annexe

8.1 Code source C++

Code source édité à l'aide d'a2ps.